

Amendment to the Claims

1. (currently amended) A computer-based method of implementing rules in a
5 rules-based system, comprising:
 - providing a first ~~rules metaphor~~ interface in the rules-based system;
 - providing a second ~~rules metaphor~~ interface in the rules-based system, the second ~~rules metaphor~~ interface different from the first ~~rules metaphor~~ interface;
 - defining a ruleflow having at least one task;
- 10 selecting either the first ~~rules metaphor~~ interface or the second ~~rules metaphor~~ interface to implement a first ruleset corresponding to the at least one task; and
implementing the first ruleset using a selected ~~rules metaphor~~ interface, the first ruleset comprising at least one rule.
- 15 2. (currently amended) The computer-based method of claim 1 in which either the first or second ~~rules metaphor~~ interfaces is a decision table metaphor.
3. (original) The computer-based method of claim 2 in which the act of implementing the ruleset using the decision table metaphor comprises:
 - 20 analyzing parameters of the task to determine information elements needed to make a decision to implement functionality of the at least one task; and
defining a grid of one or more cells corresponding to the information elements.
4. (original) The computer-based method of claim 3 in which the grid of one or more
25 cells comprises:
 - a condition cell corresponding to a condition; and
an action cell corresponding to the condition cell, the action cell comprising an action that occurs upon satisfying a condition defined for the condition cell.
- 30 5. (original) The computer-based method of claim 3 in which each cell in the grid of one or more cells has a defined object type having attributes relating to the function of the cell.

6. (original) The computer-based method of claim 5 in which a plurality of cell object types are employed in the grid of one or more cells.

5 7. (original) The computer-based method of claim 3 in which a cell in the grid of one or more cells comprises a formula.

8. (original) The computer-based method of claim 3 in which a cell in the grid of one or more cells comprises a call to an external function.

10 9. (original) The computer-based method of claim 3 in which a cell in the grid of one or more cells references an external ruleset.

10. (original) The computer-based method of claim 3 in which a cell in the grid of one or more cells references another decision table.

15 11. (original) The computer-based method of claim 3 in which the grid of one or more cells comprises a multi-column arrangement.

12. (original) The computer-based method of claim 11 in which the multi-column arrangement comprises a plurality of dimensions for cell grids in the grid of one or more cells.

13. (original) The computer-based method of claim 2 in which the decision table metaphor is an AND decision table.

25 14. (original) The computer-based method of claim 13 in which the AND decision table is organized as a traditional AND decision table.

15. (original) The computer-based method of claim 13 in which the AND decision 30 table has an inverted action section organized with a condition section.

16. (original) The computer-based method of claim 2 in which the decision table metaphor is an AND/OR decision table.

17. (original) The computer-based method of claim 2 in which the decision table metaphor is an AND decision table.

18. (currently amended) The computer-based method of claim 1 in which either the
5 first or second ~~rules metaphor~~ interfaces is a tree metaphor.

19. (original) The computer-based method of claim 18 in which the act of implementing the first ruleset using the tree metaphor comprises:

analyzing parameters of the at least one task to determine parameters to
10 implement functionality of the at least one task; and
defining a tree of one or more nodes corresponding to the parameters.

20. (original) The computer-based method of claim 19 in which the one or more nodes comprise a node having one or more successor paths based upon evaluation
15 of a condition.

21. (original) The computer-based method of claim 19 in which the one or more nodes comprise a node having one or more successor paths based upon probability factors.

20
22. (original) The computer-based method of claim 19 in which the one or more nodes comprise a node containing the conclusion of a decision making process.

23. (original) The computer-based method of claim 22 in which the node containing
25 the conclusion of a decision making process comprises an action to apply.

24. (original) The computer-based method of claim 18 in which the tree metaphor is selected from the group consisting of decision tree metaphor, classification tree metaphor, and computation tree metaphor.

30
25. (original) The computer-based method of claim 1 in which either the first or second rules metaphor interfaces is a scorecard metaphor.

26. (original) The computer-based method of claim 25 in which the act of implementing the first ruleset using the scorecard metaphor comprises:

analyzing parameters of the task to determine object attributes to implement functionality of the at least one task; and

5 defining a scorecard having point values assigned to possible object attribute values.

27. (original) The computer-based method of claim 1 in which either the first or second rules

10 metaphor interfaces is a questionnaire metaphor.

28. (original) The computer-based method of claim 27 in which the act of implementing the first

ruleset using the questionnaire metaphor comprises:

15 analyzing parameters of the task to define a set of questions to be presented; and

defining follow-up questions for responses to answers to the set of questions.

29. (original) The computer-based method of claim 27 in which the first ruleset is

20 generated based upon the defined set of question and follow-up questions.

30. (original) The computer-based method of claim 1 in which either the first or second rules

metaphor interfaces is a configuration verification metaphor.

25

31. (original) The computer-based method of claim 27 in which the act of implementing the first ruleset using the configuration verification metaphor comprises:

analyzing parameters of the task to identify a set of choice parameters

30 having

interrelationships; and

defining relationships between members in the set of choice parameters.

32. (original) The computer-based method of claim 31 further comprising:

defining individual options for the members in the set of choice parameters.

33. (original) The computer-based method of claim 31 further comprising:
defining an aggregation of options for the members in the set of choice
5 parameters.

34. (original) The computer-based method of claim 31 further comprising:
defining conflicts for the members in the set of choice parameters.

10 35. (original) The computer-based method of claim 31 further comprising:
defining mandatory relationships for the members in the set of choice
parameters.

15 36. (original) The computer-based method of claim 31 further comprising:
defining recommendations for the members in the set of choice parameters.

37. (currently amended) The computer-based method of claim 1 in which either the
first or second ~~rules~~-metaphor interfaces is a state transition diagram metaphor.

20 38. (currently amended) The computer-based method of claim 1 in which the
ruleflow comprises a second task, the method further comprising:
selecting the first ~~rules~~-metaphor interface to implement a second ruleset
corresponding to the second task if the second ~~rules~~-metaphor interface was
selected to implement the first ruleset; and

25 selecting the second ~~rules~~-metaphor interface to implement a second ruleset
corresponding to the second task if the first ~~rules~~-metaphor interface was selected to
implement the first ruleset.

30 39. (original) The computer-based method of claim 38 in which the second ruleset
utilizes the
results of the first ruleset.

40. (currently amended) A computer program product that includes a medium
usable by a processor, the medium having stored thereon a sequence of instructions

which, when executed by said processor, causes said processor to execute a process for implementing rules in a rules-based system, said process comprising:

providing a first ~~rules-metaphor~~ interface in the rules-based system;

providing a second ~~rules-metaphor~~ interface in the rules-based system, the

5 second

~~rules-metaphor~~ interface different from the first ~~rules-metaphor~~ interface;

defining a ruleflow having at least one task;

selecting either the first ~~rules-metaphor~~ interface or the second ~~rules-metaphor~~ interface to implement a first ruleset corresponding to the at least one task; and

10 implementing the first ruleset using a selected ~~rules-metaphor~~ interface, the first ruleset comprising at least one rule.

41. (currently amended) The computer program product of claim 40 in which either the first or second ~~rules-metaphor~~ interfaces is a decision table metaphor.

15

42. (original) The computer program product of claim 41 in which the act of implementing the ruleset using the decision table metaphor comprises:

20 analyzing parameters of the task to determine information elements needed to make a decision to implement functionality of the at least one task; and

defining a grid of one or more cells corresponding to the information elements.

43. (original) The computer program product of claim 42 in which the grid of one or more cells

25 comprises:

a condition cell corresponding to the condition; and

an action cell corresponding to the condition cell, the action cell comprising an action that occurs upon satisfying a condition defined for the condition cell.

30 44. (original) The computer program product of claim 42 in which each cell in the grid of one or more cells has a defined object type having attributes relating to the function of the cell.

45. (original) The computer program product of claim 44 in which a plurality of cell object types are employed in the grid of one or more cells.

46. (original) The computer program product of claim 42 in which a cell in the grid of one or more cells comprises a formula.

47. (original) The computer program product of claim 42 in which a cell in the grid of one or more cells comprises a call to an external function.

48. (original) The computer program product of claim 42 in which a cell in the grid of one or more cells references an external ruleset.

49. (original) The computer program product of claim 42 in which a cell in the grid of one or more cells references another decision table.

50. (original) The computer program product of claim 42 in which the grid of one or more cells comprises a multi-column arrangement.

51. (original) The computer program product of claim 50 in which the multi-column arrangement comprises a plurality of dimensions for cell grids in the grid of one or more cells.

52. (original) The computer program product of claim 41 in which the decision table metaphor is an AND decision table.

53. (original) The computer program product of claim 52 in which the AND decision table is organized as a traditional AND decision table.

54. (original) The computer program product of claim 52 in which the AND decision table has an inverted action section organized with a condition section.

55. (original) The computer program product of claim 41 in which the decision table metaphor is an AND/OR decision table.

56. (original) The computer program product of claim 41 in which the decision table metaphor is an AND decision table.

57. (original) The computer program product of claim 40 in which either the first or second ~~rules~~-metaphor Interfaces is a tree metaphor.

10 58. (original) The computer program product of claim 57 in which the act of implementing the first ruleset using the tree metaphor comprises:

analyzing parameters of the at least one task to determine parameters to implement functionality of the at least one task; and

defining a tree of one or more nodes corresponding to the parameters.

15

59. (original) The computer product program of claim 58 in which the one or more nodes comprise a node having one or more successor paths based upon evaluation of a condition.

20 60. (original) The computer program product of claim 58 in which the one or more nodes comprise a node having one or more successor paths based upon probability factors.

25 61. (original) The computer program product of claim 58 in which the one or more nodes comprise a node containing the conclusion of a decision making process.

62. (original) The computer program product of claim 61 in which the node containing the conclusion of a decision making process comprises an action to apply.

30

63. (original) The computer program product of claim 57 in which the tree metaphor is selected from the group consisting of decision tree metaphor, classification tree metaphor, and

computation tree metaphor.

64. (currently amended) The computer program product of claim 40 in which either the first or second ~~rules~~-metaphor interfaces is a scorecard metaphor.

5

65. (original) The computer program product of claim 64 in which the act of implementing the first ruleset using the scorecard metaphor comprises:

analyzing parameters of the task to determine object attributes to implement functionality of the at least one task; and

10 defining a scorecard having point values assigned to possible object attribute values.

66. (currently amended) The computer program product of claim 40 in which either the first or second ~~rules~~-metaphor interfaces is a questionnaire metaphor.

15

67. (original) The computer program product of claim 66 in which the act of implementing the first ruleset using the questionnaire metaphor comprises:

analyzing parameters of the task to define a set of questions to be presented; and

20 defining follow-up questions for responses to answers to the set of questions.

68. (original) The computer program product of claim 66 in which the first ruleset is generated based upon the defined set of questions and follow-up questions.

25 69. (currently amended) The computer program product of claim 40 in which either the first or second ~~rules~~-metaphor interfaces is a configuration verification metaphor.

70. (original) The computer program product of claim 69 in which the act of implementing the first ruleset using the configuration verification metaphor

30 comprises:

analyzing parameters of the task to identify a set of choice parameters having interrelationships; and

defining relationships between members in the set of choice parameters.

71. (original) The computer program product of claim 70 further comprising:
defining individual options for the members in the set of choice parameters.

72. (original) The computer program of claim 70 further comprising:
5 defining an aggregation of options for the members in the set of choice parameters.

73. (original) The computer program product of claim 70 further comprising:
defining conflicts for the members in the set of choice parameters.

10 74. (original) The computer program product of claim 70 further comprising:
defining mandatory relationships for the members in the set of choice parameters.

15 75. (original) The computer program product of claim 70 further comprising:
defining recommendations for the members in the set of choice parameters.

76. (currently amended) The computer program product of claim 40 in which either
the first or second ~~rules~~ metaphor interfaces is a state transition diagram metaphor.

20 77. (currently amended) The computer program product of claim 40 in which the
ruleflow comprises a second task, the method further comprising:

selecting the first ~~rules~~ metaphor interface to implement a second ruleset
corresponding to the second task if the second ~~rules~~ metaphor interface was

25 selected to

implement the first ruleset; and

selecting the second ~~rules~~ metaphor interface to implement a second ruleset
corresponding to the second task if the first ~~rules~~ metaphor interface was selected to
implement the first ruleset.

30

78. (original) The computer program product of claim 77 in which the second
ruleset utilizes the results of the first ruleset.